Feature detection using spikes: the greedy approach.

Laurent Perrinet
Institut de Neurosciences Cognitives de la Méditerranée
(INCM- UMR 6193, CNRS)
31, ch. Joseph Aiguier, 13402 Marseille Cedex 20, France.
Laurent.Perrinet@incm.cnrs-mrs.fr
Tel. : +33-04 91 16 45 23, Fax : +33- 04 91 77 93 04

Abstract

A goal of low-level neural processes is to build an efficient code extracting the relevant information from the sensory input. It is believed that this is implemented in cortical areas by elementary *inferential* computations dynamically extracting the most likely parameters corresponding to the sensory signal. We explore here a neuro-mimetic feed-forward model of the primary visual area (V1) solving this problem in the case where the signal may be described by a robust linear generative model. This model uses an over-complete dictionary of primitives which provides a distributed probabilistic representation of input features. Relying on an efficiency criterion, we derive an algorithm as an approximate solution which uses incremental *greedy* inference processes. This algorithm is similar to 'Matching Pursuit' and mimics the parallel architecture of neural computations. We propose here a simple implementation using a network of spiking integrate-and-fire neurons which communicate using lateral interactions. Numerical simulations show that this *Sparse Spike Coding* strategy provides an efficient model for representing visual data from a set of natural images. Even though it is simplistic, this transformation of spatial data into a spatio-temporal pattern of binary events provides an accurate description of some complex neural patterns observed in the spiking activity of biological neural networks.

Keywords: *Neuronal representation, inverse linear model, over-complete dictionaries, distributed probabilistic representation, spike-event computation, Matching Pursuit, Sparse Spike Coding.*

1 Toward a functional model of the neural code

A major problem in neuroscience is to understand the content of the activity that is observed in biological neurons. These complex activity patterns that are the basis of our cognitive abilities remain a mystery and there is yet no known unifying model explaining the "language" that could be used by neurons at the various scales of the central nervous system. In particular, descriptive models of the neural activity tend to be incomplete or to reflect a distorted description of natural conditions [18]. We will try here to overcome these problems by precisely defining the model and the hypotheses that we want to validate. We will assume here that there exists a functional *neural code*

and that we may decipher the neural activity by exploring algorithms —based on the nature and architecture of the neural system— that solve efficiently the function that is provided by the system. We will illustrate this method for the primary visual area (V1) in the human by trying to define precisely its function and then by proposing a model for the neuronal representation and for the mechanisms that may implement it.

1.1 Solving inverse problems using neural networks

V1 is a cortical area specialized in low-level visual processing from which the majority of the visual information diverges to higher visual areas. We will describe it here as implementing an inverse problem by *analyzing* images thanks to an internal model. The hypothesized function over the long term (in the order of hours to years) will thus be to process natural scenes (that is images that occur frequently) so as to progressively build a "model" of their structure. The goal is that for any of these images, this model must rapidly (in the order of a fraction of a second) represent a set of features relevant to that image[1] and corresponding to this model (see Fig. 1). This representation, including for instance the location and orientation of the edges that outline the shape of an object, is then relayed to higher level areas to allow, for instance, a robust recognition of useful patterns. Actually, this is similar to numerous tasks in engineering and applied mathematics, where a reverse-engineering process allows to find a robust representation of the data (such as an estimation of the internal state of a system in control theory) by identifying the so-called *hidden parameters* of the system. The success of this algorithm over the long term (in the order of days to generations) allows then to validate the model that was learned through the pressure of evolution. In this framework, it is thus easier to describe cortical activity as the result of the inversion (or analysis) of an internal model of the world.

Moreover, such a model of the world should also take into account some basic knowledge of actual physical interactions. This idea is based on the assumption that the observations are the effect of the interplay between different causes corresponding to stable physical interactions and that they should be recovered to describe the observed data by representing the underlying actual physical structure. In particular, some knowledge of the usual transforms of the signal (such as translation and scaling in images) which are related to regularly occurring changes in the physical world (lateral and frontal translations of objects in space) allows then for a robust representation and further analysis by higher level cortical areas. This may finally allow for desired properties such as an invariant representation of objects to these frequently occurring transformations.

We will restrict here this artificial neural network to a feed-forward model of V1 which processes flashed static images[2]. We will assume that the model of natural images is fixed and accurate and we will define the goal of our model as recovering the sources (corresponding to some hidden state variables, see Fig. 1) from an observed static image. Moreover, in the framework of natural living systems, we will assume that a main constrain from evolution is the ability to process the information as quickly as possible. This model will consist in these restrictive conditions to a one-layered neural network as illustrated in Fig. 2 and the output of the neural layer should describe at best and as

[1] We will consider here that each neuron may be characterized by a preferred pattern to represent. It should though be emphasized that this view differs from the "grand-mother neuron" paradigm since the representation emerges from the interaction of different active neurons.

[2] In particular, we will study the transient response of the network and neglect the information fed back by higher areas. This latter information will be necessary in more complex algorithms which take into account the context of a local feature.

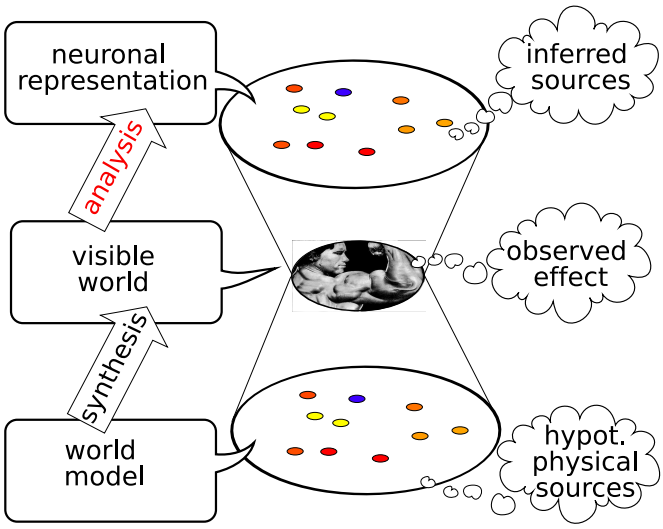

Figure 1: **Inverse-mapping as a a goal for sensory neural coding**. The visible world is modeled as the interaction of a large set of hypothetical physical sources (world model) according to a known model of their interactions ("synthesis"). We will consider that for sensory cortical areas, the goal of the neural representation (and its implementation by the *neural code*) is to analyze the signal so as to recover at best and as quickly as possible the sources that generated the signal ("analysis") . The analysis may thus be considered as an inverse mapping of the synthesis. A proposed solution for this problem is to *infer* at best the most probable hidden state.

quickly as possible the visual content. Considering the system as an information channel (according to the definition of Shannon and Weaver [27]) which processes samples from the set of natural images, we may therefore define the goal of V1 as to transmit the information about the sources (in *bits*) at the highest rate as possible.

1.2 Inverse models for sensory processing

To build an algorithm of the inverse model to efficiently code the input, we will first define the forward synthesis model as a *Linear Generative Model* (LGM) as is often assumed for natural images [17]. For visual data, images consist indeed of the set of observed luminance values from different spatial positions and a fairly good approximation —especially for small images of non-occluding objects— considers the image as the linear combination of "primitive images", similarly to the superposition of transparent layers. This approximation is based on the assumptions that the energy of the photonic flow from a spatial position (the luminosity) consists of the multiplicative interaction of different "shapes" that contribute each for a fraction of the global luminosity. Thanks to the non-linear *gamma* transform of luminosities into luminances [22] which approaches a logarithmic function, these "shapes" add up linearly in the luminance space. Although this is justified in practice for transparent shapes, it is not for occlusions. The LGM framework provides however a general framework for describing natural images.
The forward model defines images as the superposition of shapes of different intensi-

ties which correspond in our framework to scalar "hidden states". Formally, we will describe this set of scalars by a vector[3] $\mathbf{s} = \{s_j\}_{1 \leq j \leq N}$ where N is the dimension of the dictionary. Similarly, one image will be described as a point in a multidimensional state space of dimension M where every pixel corresponds to one dimension (and therefore the pixel value will be its scalar value along this dimension). This observation signal will be written $\mathbf{x} = \{x_i\}_{1 \leq i \leq M}$ over the set of spatial positions denoted by their address i (that is the pixels over a rectangular grid in an image processing framework). To define the LGM, we will use a "dictionary" of images as the matrix $\mathbf{A} = \{\mathbf{A}_j; 1 \leq j \leq N\}$ of the N images of the "primitive shapes" $\mathbf{A}_j = \{A_{ij}\}_{1 \leq i \leq M}$. The image corresponding to the internal state $\mathbf{s}$ will finally be defined as:

$$\mathbf{x} = \sum\nolimits_{1 \leq j \leq N} s_j . \mathbf{A}_j \tag{1}$$

This model of natural images is defined by the statistics of the sources $\mathcal{S}$ and by the dictionary $\mathbf{A}$ of primitive images. The latter corresponds to the set of basis functions which describe the space of all observed natural images $\mathcal{I} = \{\mathbf{x}\}$ that we wish to characterize.
In this paper, we will use the same fixed dictionary of filters (that is $\mathbf{A}$) and assume similar hypotheses on the statistics of $\mathcal{S}$ to rate the efficiency of different coding strategies. Using this formalization, the function of the neural network consists in recovering the sources by inverting the synthesis process. The results of this inversion (in the space of the neural representation) will thus share the same dimension (that we noted N) as the space of the sources, that is the cardinal of the dictionary. As a first approximation (and as is observed in simple cells from V1), the dictionary of primitive shapes will correspond to localized orientation selective edges at different positions and scales resembling Gabor functions [11, 23] at different spatial scales. This may be particularly adapted in an information theoretic based framework as these shapes correspond to independent features in natural scenes [4]. We choose here that the forward model will be described by a wavelet transform [14] and we will use this architecture to compare different coding strategies.

1.3 Efficient coding of natural images

In fact, particular care should be put on the parameters of this wavelet architecture. In particular, it is desirable for the representations of natural images to be robust to natural conditions. As is the case for natural images, we will consider that the observed signals are generated by sources that share certain features which differ by continuous transformations such as edges at different time, position, orientation or scale. Since the corresponding spatial transformations (translations, rotations and scaling) are very common, if there exists a corresponding transformation in the source space (that is if this transformation of all sources are in the dictionary), the resulting representation of the transformed image should simply be derived by a transformation (in the source space) of the original representation. Thus, it is necessary for the dictionary to be invariant according to these usual transformations for the representation to be robust. In particular, this allows for instance for higher level areas to detect some specific inputs with an invariance to usual transformations. Typically, this robustness constraint implies in our architecture that the tiling of the wavelet filters is smoother than an orthogonal representation [21]. As a consequence, the dictionary will be over-complete,

[3] in the following, we will denote vectors and matrices by bold characters

i.e. the number of dictionary elements will be of several orders of magnitude larger than the dimension of the image space (that is $N >> M$).
From the definition of the forward model, for any signal $\mathbf{x}$, there exist at least one set of parameters $\mathbf{s}$ which recovers the observed signal. However, in the case where the dictionary is over-complete, the inversion of the LGM will not yield an unique solution in $\mathcal{S}$ to any given signal in $\mathcal{I}$: the problem is ill-posed. The coding strategies corresponding to possible 'analysis' algorithms (see Fig. 1) have different efficiencies and, in particular, the solution given by the wavelet coefficients with an over-complete dictionary yields an highly redundant representation. According to Barlow [3], the goal of sensory processing would be rather to choose the most efficient representation: following the same argument as the Occam razor, whenever there is the choice between two representations, the best is the one that is the most parsimonious. In our framework, a possible goal would be to maximize the mean codeword length, that is get the coding strategy that describes at best the images. From Shannon's source coding theorem [27], this length is bounded by the entropy of the images for a given architecture and coding strategy. Under some assumptions that we will develop later, this is equivalent to find the *sparsest representation*, that is the representation that uses the smallest number of sources [17]. This sparseness constraint thus allows to restrict the different solutions of the inversion of the forward model so as to find an appropriate candidate for the neural code.
However, the combinatorial complexity of the inverse problem grows very quickly as the dimension of the dictionary increases (it's NP-complete, see [14]). There exists therefore no simple algorithm that optimizes exactly the problem in reasonable time as we handle more complex signals such as natural images, but acceptable sub-optimal strategies to approach this problem do exist (see a review in [19]). Most popular solutions optimize a compromise between the reconstruction error and the sparsity and are based on linear optimization or gradient approaches [28]. Following the same arguments as Barlow [3], we explore an alternate solution which uses a probabilistic representation and Bayesian inference.

2 Sparse spike coding using a greedy inference pursuit

Focusing on the event-based nature of axonal information transduction and in order to reflect the parallel architecture of the nervous system, we will here propose a solution for inverting the forward model that we defined for natural images. This will build a Bayesian inference framework based on feature-matching neurons and on spikes as events representing primitive "decisions".

2.1 Greedy inference pursuit using spikes

This approach proposes an alternative to classical paradigms of neural coding such as the spike-rate coding approach of the *perceptron* (see Fig. 2). Instead of coding information in the mean firing frequency of neurons, we will present an original approach solving the function that we defined above. It uses a distributed probabilistic representation and we will assume here that the activity of neurons (such as the membrane potential) in the layer will represent dynamically the evidence of a correct match and that the output spiking signal signifies a set of elementary decisions made by the neurons. Following this process and focusing on every single spike, the process occurs repeatedly using two steps: Matching (M) and Pursuit (P).

(*M*) To each neuron is assigned a vector (or weight pattern) corresponding to its preferred stimulus. Neurons compete in parallel to find the most probable *single source* component by integrating evidence according to their weight patterns. The first source to be detected should be the one corresponding to the highest activity.

(*P*) The best match is assigned a decision which, once it has been taken, is assumed to be reliable: we can take into account this information before performing any further computations (and in particular finding a new match) so as to yield a new representation where we removed *completely* the detected source.

We call this approach a *greedy pursuit* which is based on the recursion of two greedy mechanisms (detection - removal). These are here idealized but correspond to known aspects of neural activity (matching - suppression).
We will see that this method is similar to the approach developed in the method of Matching Pursuit [15]. However, instead of a heuristic scheme, the algorithm will here be derived from known hypotheses and thanks to the description of the successive steps that may lead to the greedy pursuit, it may be considered as an optimization strategy of the goal that we defined above (namely maximizing the transfer of information). We will then propose an implementation using Integrate-and-fire neurons and test the efficiency of this artificial neural code.

2.1.1 Matching: Detection of the most probable source component

First, given the signal $\mathbf{x} \in \mathcal{I}$, we are searching for the *single* source $s^*.\mathbf{A}_{j^*} \in \mathcal{I}$ that corresponds to the maximum *a posteriori* (MAP) realization for $\mathbf{x}$ (and knowing it is a realization of the LGM as it is defined in Eq. 1). We will address in general a single source by its index and strength by $\{j, s\}$ so that the corresponding vector in $\mathcal{S}$ corresponds to a vector of zero values except for the value s at index j. The MAP is defined by:

$$\{j^*, s^*\} = \text{ArgMax}_{\{j,s\}} P(\{j, s\}|\mathbf{x}) \tag{2}$$

To evaluate $P(\{j, s\}|\mathbf{x})$, the probability *a posteriori* of a single source knowing the signal, we have from Bayes' theorem

$$\{j^*, s^*\} = \text{ArgMax}_{\{j,s\}} [P(\mathbf{x}|\{j, s\}).P(\{j, s\})] \tag{3}$$

where $P(\mathbf{x}|\{j, s\})$ is the likelihood probability of a signal knowing a single source and $P(\{j, s\})$ is the *a priori* probability of the sources.
To compute the likelihood we have to first define the model of the measurement [13, p.26]. We will first assume that we are in a low-noise limit environment (the global contrast is optimal and the eye/camera is adapted to the scene) so that we have no or little measurement noise. Knowing one component $\{j, s\}$, the only "noise" from the viewpoint of neuron j is the combination of the unknown sources $\{\alpha_k\}_{1 \leq j \leq N}$. It is thus the residual of the signal knowing $\{j, s\}$. We may thus write the noise as

$$\mathbf{x} = s.\mathbf{A}_j + \nu \text{ with } \nu = \sum\nolimits_k \alpha_k.\mathbf{A}_k \tag{4}$$

The residual of the signal (an image) is thus considered as an undetermined perturbation[4]. Assuming that the α_k are independent random variables (since we know only

[4]It should be stressed that the image model is still deterministic.

$\{j, s\}$), from the central limit theorem it comes that for a sufficiently high number of sources, the distribution of the random variable ν converges to a normal distribution with known mean and covariance matrix. From the work of Atick [1], we know that for natural images this normal distribution is fairly homogeneous across natural images. We may either use another metric (based on the Mahalanobis distance, as exposed in [21]) or use a decorrelating kernel to yield a spherical probability distribution centered around the origin ($E(\nu) = 0$) of this "noise". Normalizing by the mean energy of images in $\mathcal{I}$, the residual signal is thus considered as a decorrelated noise of unit variance. From $P(\mathbf{x}|\{j, s\}) = P(\mathbf{x} - s.\mathbf{A}_j) = P(\nu)$, it follows

$$\begin{aligned} \{j^*, s^*\} &= \text{ArgMax}_{\{j,s\}}[\log P(\mathbf{x}|\{j, s\}) + \log P(\{j, s\})] \\ &= \text{ArgMin}_{\{j,s\}}[\|\mathbf{x} - s.\mathbf{A}_j\|^2/2 - \log P(\{j, s\})] \end{aligned} \quad (5)$$

We will further consider that the dictionary was learned so that over a long period the neurons have similar statistics: the prior is uniform across sources and values. We thus have no prior knowledge or preference for any source. It thus comes

$$\begin{aligned} \{j^*, s^*\} &= \text{ArgMin}_{\{j,s\}}\|\mathbf{x} - s.\mathbf{A}_j\|^2 \\ &= \text{ArgMin}_{\{j,s\}}[s^2.\|\mathbf{A}_j\|^2 - 2.s. < \mathbf{x}, \mathbf{A}_j >] \end{aligned}$$

To minimize this bi-variate function, we may first minimize for every element j the coefficient s_j to get the corresponding $s_j^* = \text{ArgMax}_s P(\{j, s\}|\mathbf{x})$. From the above equations, this is equivalent to minimizing in the last equation the quadratic function of s which is minimal for the scalar coefficient

$$s_j^* = \frac{< \mathbf{x}, \mathbf{A}_j >}{\|\mathbf{A}_j\|^2} \quad (6)$$

that is for the scalar projection of the input on $\mathbf{A}_j$. Then, since for every element j, $s_j^*.\mathbf{A}_j$ is the projection of $\mathbf{x}$ on $\mathbf{A}_j$, so that $s_j^*.\mathbf{A}_j$ and $\mathbf{x} - s_j^*.\mathbf{A}_j$ are orthogonal, it follows from Pythagoras's theorem

$$\begin{aligned} j^* &= \text{ArgMin}_j[\|\mathbf{x} - s_j^*.\mathbf{A}_j\|^2] \\ &= \text{ArgMin}_j[\|\mathbf{x}\|^2 - \|\frac{< \mathbf{x}, \mathbf{A}_j >}{\|\mathbf{A}_j\|^2}.\mathbf{A}_j\|^2] = \text{ArgMax}_j\|\frac{< \mathbf{x}, \mathbf{A}_j >}{\|\mathbf{A}_j\|}\|^2 \\ j^* &= \text{ArgMax}_j|< \mathbf{x}, \frac{\mathbf{A}_j}{\|\mathbf{A}_j\|} >| \end{aligned} \quad (7)$$

Finally, as defined in Eq. 2, we found that the source component that maximizes the probability is the projection of the signal on the normalized elements of the dictionary. This justifies the computation of the correlation in the perceptron model [25] as it provides a measure of the log-probability under the assumptions that we used. However, using a different strategy as these linear systems, we will associate in our greedy approach this inference with a lateral propagation of this information to the correlated neurons and only then resume the algorithm.

2.1.2 Pursuit: Lateral interaction and Greedy pursuit of the best components

Before detecting another single source component, we will take into account the information that we extracted from the signal by propagating it to the neighboring neurons using lateral interaction links. As we found the MAP source knowing the signal $\mathbf{x}$, we

may pursue the algorithm by accounting for this inference on the signal knowing the element that we found. From

$$P(\{j,s\}|\mathbf{x},\{j^*,s^*\}) = P(\{j,s\}|\mathbf{x} - s^*.\mathbf{A}_{j^*}) \tag{8}$$

and since source are here supposed to have independent activities[5], the pursuit algorithm assumes that —knowing the previous detection— we may resume the detection on this residual signal. We will thus use this new residual signal in which we will then find a new component corresponding to the most probable single source.
In this recursive approach, we will note as n the rank of the step in the pursuit (which begins at $n = 0$ for the initialization). Writing $N_j = \|A_j\|$, the first scalar projection that we have to maximize —and which will serve as the initialization of the algorithm—is given by :

$$C_j^{(0)} =< \mathbf{x}, \frac{\mathbf{A}_j}{N_j} > \tag{9}$$

Let's also note the address of the successive winning neuron from the first step $n = 1$ as

$$j^{(n)} = \text{ArgMax}_j |C_j^{(n-1)}| \tag{10}$$

Knowing $j^{(n)}$, in order to resume the pursuit at the next step, we saw that we need to compute the projection of the signal on the elements of the dictionary. Let's therefore set initially $\mathbf{x}^{(0)} = \mathbf{x}$ and $\mathbf{x}^{(n)}$ the successive residuals. In this greedy approach, we consider that the decision corresponding to the MAP criteria at step n is correct and that we may therefore update the residual and the corresponding activities $C_j^{(n-1)} =< \mathbf{x}^{(n-1)}, \frac{\mathbf{A}_j}{N_j} >$ by subtracting to $\mathbf{x}^{(n-1)}$ its projection on the winning element of index $j^{(n)}$ (see Eq. 6) :

$$\mathbf{x}^{(n)} = \mathbf{x}^{(n-1)} - C_{j^{(n)}}^{(n-1)}.\frac{\mathbf{A}_{j^{(n)}}}{N_{j^{(n)}}} \tag{11}$$

Furthermore, we don't need to feed this information back to the signal and we may directly compute the activity again for all vectors thanks to the linearity of the scalar product operator:

$$\begin{aligned} C_j^{(n)} &= < \mathbf{x}^{(n)}, \frac{\mathbf{A}_j}{N_j} > \\ &= < \mathbf{x}^{(n-1)} - C_{j^{(n)}}^{(n-1)}.\frac{\mathbf{A}_{j^{(n)}}}{N_{j^{(n)}}}, \frac{\mathbf{A}_j}{N_j} > \\ C_j^{(n)} &= C_j^{(n-1)} - C_{j^{(n)}}^{(n-1)}. < \frac{\mathbf{A}_j}{N_j}, \frac{\mathbf{A}_{j^{(n)}}}{N_{j^{(n)}}} > \end{aligned} \tag{12}$$

In this simplified framework, the choice of the best match and the update rule are independent of the choice of the norm N_j of the filters (see Eq. 10 and 12), so that we may indifferently use in the following normalized filters (that is $N_j = 1$ for all neurons) so as to simplify the equations. It comes thus:

$$\text{(Initialization)} \quad \boxed{C_j^{(0)} =< \mathbf{x}, \mathbf{A}_j >} \tag{13}$$

[5]For any realization of the images, individual sources have independent activities, that is that removing one source, one gets a new image (conform with the LGM model) and one does not change the probability distribution of the other sources.

This activities' update (Eq. 12) corresponds in neuro-physiological terminology to a lateral interaction. It will be proportional to $R_{j,j^{(n)}}$ where $R_{j,j^{(n)}} =< \mathbf{A}_j, \mathbf{A}_{j^{(n)}} >$ is the correlation of any element j with the winning element $j^{(n)}$ and relates to the reproducing kernel in wavelet theory.
Finally, we achieve the recursive greedy pursuit of best components as the iteration of respectively a "matching" and a "pursuit" step. While the residual energy is greater than a fixed threshold $\|\mathbf{x}^{(n)}\| > \varepsilon$, we compute :

$$\text{(Matching)} \quad \boxed{j^{(n)} = \text{ArgMax}_j |C_j^{(n-1)}|} \tag{14}$$

$$\text{(Pursuit)} \quad \boxed{C_j^{(n)} = C_j^{(n-1)} - C_{j^{(n)}}^{(n-1)}.R_{j,j^{(n)}}} \tag{15}$$

The greedy pursuit therefore transforms an incoming signal $\mathbf{x}$ in a list of ranked sources $\{j^{(n)}, s^{(n)}\}$ such that finally (from Eq. 11) the signal may be reconstructed as

$$\mathbf{x} = \sum\nolimits_{k=1\ldots n} s^{(k)}.\mathbf{A}_{j^{(k)}} + \mathbf{x}^{(n)}$$

which is an approximation of the goal set in inverting Eq. 1 if the norm of the residual signal $\mathbf{x}^{(n)}$ converges to zero.

2.1.3 Properties of the greedy pursuit

This algorithm is exactly equivalent to Matching Pursuit [15]. This algorithm is familiar in signal processing and is increasingly used for image and video processing [9, 6]. However, the use of the statistics of natural images statistically optimizes the coding efficiency by modifying the image space metric [21]. Moreover, the Bayesian inference framework allows to precisely tune the heuristic approach of the Matching Pursuit. It allows for instance to set a different prior or to include knowledge of the measurement noise that is adapted to the goal of the system (and hence a different matching criteria that may depend on the N_j). This algorithm presents similar computational complexity and properties [14, pp.412–9]. In particular

$$C_{j^{(n)}}^{(n)} = C_{j^{(n)}}^{(n-1)} - C_{j^{(n)}}^{(n-1)} = 0 \tag{16}$$

and as a consequence the activity of a winning neuron is totally canceled.
Moreover, although filters in the dictionary are here generally not orthogonal, the residual image is orthogonal to the winning filter and

$$\|\mathbf{x}^{(n)}\|^2 = \|\mathbf{x}^{(n-1)}\|^2 - |s^{(n)}|^2.\|\mathbf{A}_{j^{(n)}}\|^2 \tag{17}$$

so that we may easily compute the Squared Error (SE) of the residual signal at every step of the coding.

$$\begin{aligned} \mathtt{SE}^{(n)} &= \|\mathbf{x} - \sum\nolimits_{k=1\ldots n} s^{(k)}.\mathbf{A}_{j^{(k)}}\|^2 = \|\mathbf{x}^{(n)}\|^2 \\ &= \mathtt{SE}^{(n-1)} - |s^{(n)}|^2.\|\mathbf{A}_{j^{(n)}}\|^2 \qquad (18) \\ \mathtt{SE}^{(n)} &= \|\mathbf{x}\|^2 - \sum\nolimits_{k=1\ldots n} |s^{(k)}|^2.\|\mathbf{A}_{j^{(k)}}\|^2 \\ &= \|\mathbf{x}\|^2 - \sum\nolimits_{k=1\ldots n} |C_{j^{(k)}}^{(k-1)}|^2 \qquad (19) \end{aligned}$$

It first implies that the stopping criteria may be computed using this computation without computing $\|\mathbf{x}^{(n)}\|$. A further consequence of the monotonous decrease of the SE

from Eq. 19 is —under the condition that the dictionary is at least complete— the convergence of the reconstruction [14, p.414]. Under this condition, the algorithm will therefore stop in finite time.
Though simple, the greedy pursuit is a complex non-linear algorithm. In fact, the study of its behavior is non trivial and may involve chaotic dynamics [8]. In particular, it is obvious that the choice that is made at a giving step may influence all future steps. This implies that a failed match may propagate wrong information to following steps and therefore that the probability of a failure grows higher as the rank increases. These properties are discussed in [21] and in particular we illustrated that the speed of convergence increases as the dictionary becomes more over-complete so that it provides an efficient representation for natural scenes in image processing tasks.

2.2 Implementation using Integrate-and-Fire (IF) neurons

From our knowledge of neural mechanisms in a neuronal layer, the model of greedy feature pursuit that we derived from an event-based computation in a parallel architecture is particularly adapted to a model of neural computations. We will derive an implementation using a network of spiking neurons based on the same feed-forward architecture of the perceptron (see Fig. 2) but implementing the greedy pursuit using *lateral interactions*.
The activity is represented by a driving current that drives the potential V_j of Integrate-and-Fire neurons [12]. For illustration purposes, the dynamics of the neurons will here be modeled by a simple linear integration of the driving current C_j (other integration schemes lead to similar formulations):

$$\tau.\frac{d}{dt}V_j = p_j.C_j \tag{20}$$

The neurons are duplicated with opposite polarity $p_j = \pm 1$ so that $C_j = p_j.|C_j|$ to model the ON / OFF symmetry of simple cells [23]. The neuron will generate a spike when the potential reaches an arbitrary threshold that we set here to 1.

To implement the computation of the match of an input with stored patterns, we define a dictionary which will be implemented by weight vectors $\mathbf{A}_j$. These vectors are normalized as described above and the input is decorrelated. The linear feed-forward perceptron integrates synaptically the input $\mathbf{x}$ into an initial activity C_j such that

$$C_j = < \mathbf{x}, \mathbf{A}_j > \tag{21}$$

The scalar projection will therefore drive the potential of the neuron. We may predict from the monotonous integration that the first neuron to generate a spike will be the one that corresponds to the maximal rectified scalar projection of the input signal with the weight vectors of the network, that is

$$j^* = \text{ArgMax}_j |C_j| \tag{22}$$

the firing time is $t^* = \frac{\tau}{|C_{j^*}|}$ and the potential is then $V_j = \frac{t^*}{\tau}.C_j = \frac{C_j}{|C_{j^*}|}$. This is therefore a simple and biologically plausible implementation of a MAP estimate using the parallel architecture of the network which is in contrast with the complexity of this implementation on a single-processor computer. To implement the greedy algorithm, we then need to implement a lateral interaction on the neighboring neuron similar to the observed lateral propagation of information in V1 [10, 2]. In our scheme the interaction

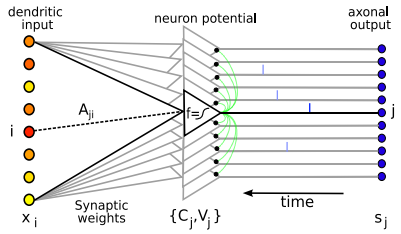

Figure 2: **Model of a neuronal layer as a communication channel**. To understand the content of neural activity, we consider here that the neuronal layer implements the inverse of a forward model (that is the analysis in Fig. 1). The architecture is similar to the perceptron: the input (noted x_i) is matched with normalized weight patterns A_{ji} (which are fixed in this paper) so as to provide an integrative activation value (the membrane potential) which in turn is non-linearly transformed to achieve a membrane potential which grows proportionally to the probability of matching a feature. Spikes represent decisions that are fed back on the correlated neighboring neurons using lateral interactions (that we represented for the first spiking neuron) but also on the axonal output which yield a spiking output s_j.

should yield the same configuration in the network (activity and potential) as if the source that was detected was originally absent from the signal. In this model, if j^* is the winning neuron, the activity should be subtracted by $|C_{j^*}|.R_{\{j,j^*\}}$ (see Eq. 15) and the potential by this value integrated over t^*. The lateral interaction is thus achieved by updating after each spike the activity of the neighboring neurons proportionally to their cross-correlation $R_{\{j,j^*\}}$ with the corresponding winning neuron j^* :

$$C_j \leftarrow C_j - |C_{j^*}|.R_{\{j,j^*\}} \tag{23}$$

and removing the potential that would be generated by the activity of the removed source:

$$V_j \leftarrow V_j - \frac{t^*}{\tau}.|C_{j^*}|.R_{\{j,j^*\}}$$

that is simply

$$V_j \leftarrow V_j - R_{\{j,j^*\}} \tag{24}$$

This lateral interaction is here immediate and behaves as a refractory period on the winning neuron ($C_{j^*} \leftarrow 0$ and $V_{j^*} \leftarrow 0$) and a graded inhibition on positively correlated neurons. It involves a subtractive hyper-polarizing term on the potential and on the activity. Biologically, it is improbable that the lateral interaction could be instantaneous, but this lateral interaction could be implemented in a fast manner using a shunting lateral interaction [5] mediated by fast-spike inter-neurons. Finally, this simple implementation therefore implements the Matching Pursuit algorithm that we defined in Eq. 14 and 15 and we will apply it to simple visual tasks.

3 Results: efficiency of Sparse Spike Coding

3.1 Coding natural image patches

We compared the method we described in this paper with similar techniques used to yield sparse and efficient codes such as the conjugate gradient method used by Ol-

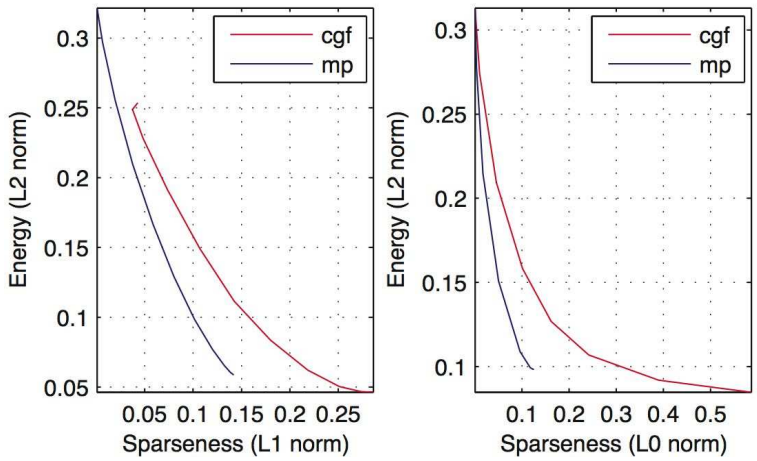

Figure 3: **Efficiency of the matching pursuit compared to conjugate gradient**. We compared here the matching pursuit ('mp') method with the classical conjugate gradient function ('cgf') method as is used in [17]. We present the results for the coding of a set of image patches drawn from a database of natural images. These results were obtained with the same fixed dictionary of edges for both methods. We plot the mean final residual error for two definitions of sparseness : **(Left)** the mean absolute sum of the coefficients and **(Right)** the number of active (or non-zero) coefficients (the coding step for MP). For this architecture, the sparse spike coding scheme appears to be more efficient to code natural image patches.

shausen and Field [16]. We used a similar context and architecture as these experiments and used in particular the database of inputs and the dictionary of filters learned in the SPARSENET algorithm. Namely, we used a set of 10^5 10×10 patches (so that $M = 100$) from whitened images drawn from a database of natural images. The weight matrix was computed using the SPARSENET algorithm with a 2-fold over-completeness ($N = 200$) that show similar structure as the receptive of simple cells in V1 [23]. From the relation between the likelihood of having recovered the signal and the squared error in the new metric, the mean squared reconstruction error (L2-norm) is an appropriate measure of the coding efficiency for these whitened images. This measure represents the mean accuracy (in terms of the logarithm of a probability) between the data and the representation. We compared here this measure for different definitions and values for the "sparseness".

First, by changing an internal parameter tuning the compromise between reconstruction error and sparsity (namely the estimated variance of the noise for the conjugate gradient method and the stopping criteria in the pursuit), one could yield different mean residual error with different mean absolute value of the coefficients (see Fig. 3, left) or L1-norm. In a second experiment, we compared the efficiency of the greedy pursuit while varying the number of active coefficients (the L0-norm), that is the rank of the pursuit. To compare this method with the conjugate gradient, a first pass of the latter method was assigning for a fixed number of active coefficients the best neurons while a second pass optimized the coefficients for this set of "active" vectors (see Fig. 3, right). Computationally, the complexity of the algorithms and the time required by both methods was similar. However, the pursuit is by construction more adapted to provide a

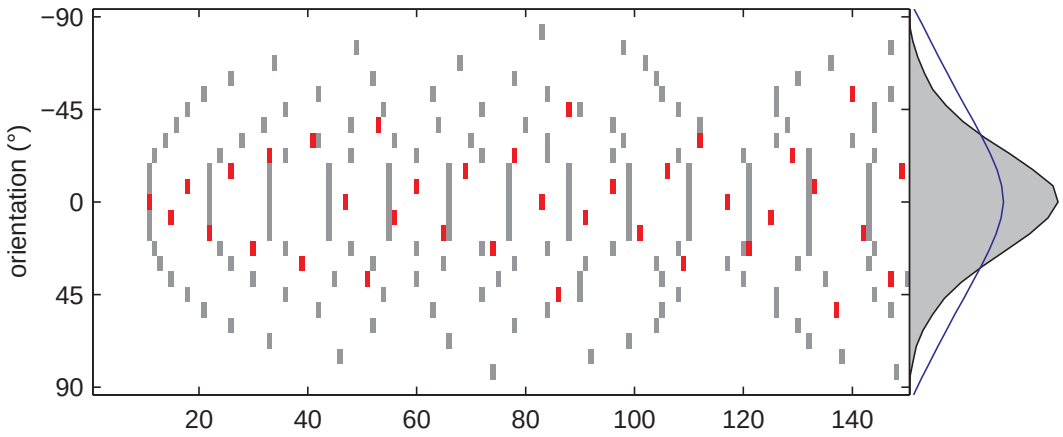

Figure 4: **Implementation of the greedy pursuit using Integrate-and-Fire Neurons**. We simulated here the activity of a network of Integrate-and-Fire neurons tuned to form a simple model of an hyper-column in the primary visual area (V1) to the presentation of a horizontal edge at $t = 0$. We show in this image the output spiking activity of 16 neurons tuned for different orientations for the feed-forward (black bars) and the sparse spike coding (white bars) models during the first 150 ms. In this latter model, the correlation linked to the information already detected is propagated as a hyper-polarizing and shunting lateral interaction to the neighboring neurons : the response in both latency and spiking frequency to the oriented edge is clearly more selective.

progressive and dynamical result while the conjugate gradient method had to be re-computed for every set of parameter. Best results are those giving a lower error for a given sparsity or a lower sparseness (better compression) for the same error. In both cases, the Sparse Spike Coding provides a coding paradigm which is of better efficiency as the conjugate gradient.

3.2 Model of a hyper-column in the primary visual area

To illustrate the properties of the algorithm, I modeled a network of linear Integrate-and-Fire neurons forming a simple model of an hyper-column in the primary visual area (V1). This model consist of an isolated network of 16 neurons selective to different orientations of contours and which are modeled as Gabor filters (which are here symmetric with circular envelopes). We compared a pure feed-forward model to a network implementing the lateral interactions that we described above (see Eq. 23 and 24). We show here the resulting spiking activity when one of the preferred stimuli (the horizontal edge) was continuously presented from time $t = 0$ (see Fig. 4).
We observe that the neuron corresponding to that preferred stimulus fires with the shortest latency but also produces the highest spike rate. Moreover, the activity of the neurons corresponding to non-preferred directions shows a lower spiking activity when implementing the greedy pursuit. This dynamic reflects the lateral interaction (here an inhibition to the positively correlated neurons) generated at every spike which is observed in V1 [7]. In fact, compared to the linear model, the latency and the frequency of the neighboring neurons show a sharper response for neighboring edge orientations (see Fig. 5) which corresponds to the high selectivity observed in simple cells from V1

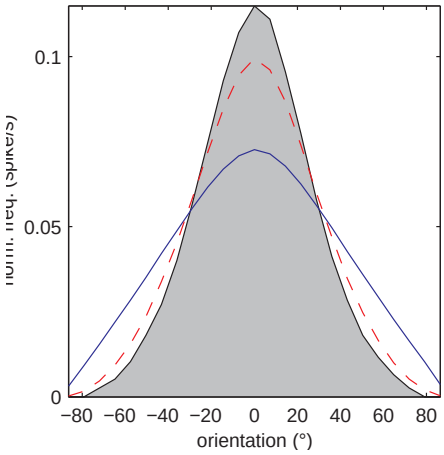

Figure 5: **Selectivity response of the network to orientation**. Output spike firing rate to the presentation of a horizontal edge at time $t = 0$. for the linear feed-forward model (plain line), the sparse spike coding scheme (filled curve) and with divisive normalization (dashed line) for different orientations of the input stimulus. The narrower tuning curve for the latter two methods represents a more selective response to the features learned in synaptic weights and mimics the behavior of the neural response in the primary visual area.

[24]. The selectivity of this model was compared with the model of *divisive normalization* [26], suggesting that this simple implementation of Integrate-and-Fire neurons — linked by lateral interactions and removing dynamically the redundancy in the signal— could provide a model for the complex processing occurring in cortical areas.

Conclusion

We presented here a model for neural processing which provides an alternative to the feed-forward and spike-rate coding approaches. Focusing on the parallel architecture of cortical areas, we based our computations on spiking events. Defining the function of sensory areas as matching the input to a model with unknown parameters, the activity of the network represented a probabilistic evaluation of the accuracy of the match. From this representation, we inferred the best match using the Bayes rule and an inference decision criterion. We then derived an algorithm which may be implemented using lateral interactions : it removes for every spike the corresponding activity to correlated neurons. Simulations of this model compare to the non-linear behavior of neurons in biological network such as the primary visual cortex (V1).
This model is based on the Matching Pursuit algorithm and provides a general framework for modeling the complex behavior of networks of spiking neurons. Particularly, it can be extended to multi-layered networks and provides an efficient code for natural images as we described elsewhere [21]. Further studies provided a learning scheme based on an Hebbian learning rule which yields an unsupervised learning of the sources as independent components of the signal to describe [20]. The model thus provides an algorithm of *Sparse Spike Coding* which is particularly efficient for visual tasks.
This simple strategy thus suggest that the inherent complexity of the neural activity

is perhaps not simply the reflection of the computational details of neurons but may rather be the consequence of the parallel event-based dynamics of the neural activity. Although our model is a simplistic caricature compared to the behavior of biological neurons, it provides a simple algorithm which is compatible with some complex characteristic of the response of neuronal populations. It thus proposes a challenge for discovering the mechanisms underlying the efficiency of nervous systems by focusing on large-scale networks of spiking neurons.

Reproducible research

Scripts reproducing all figures may be obtained from the author upon request.

Acknowledgments

The author thanks the team at the Redwood Neuroscience Institute for stimulating discussions and particularly Jeff Hawkins, Bruno Olshausen, Fritz Sommer, Tony Bell, Dileep George, Kilian Koepsell and Matthias Bethge. This work was supported by a grant form the French Research Council (Action Concertée Incitative / Temps et Cerveau).

References

[1] J. Atick. Could information theory provide an ecological theory of sensory processing? *Neural Computation*, 3(2):213–52, 1992.

[2] W. Bair, J. Cavanaugh, and J. Movshon. Time course and time–distance relationships for surround suppression in macaque V1 neurons. *The Journal of Neuroscience*, 23(20):7690 —701, August 2003.

[3] H. Barlow. Redundancy reduction revisited. *Network: Computations in Neural Systems*, 12:241—25, 2001.

[4] A. Bell and T. Sejnowski. The 'independent components' of natural scenes are edge filters. *Vision Research*, 37(23):3327–38, 1997.

[5] L.J. Borg-Graham, C. Monier, and Y. Fregnac. Visual input evokes transient and strong shunting inhibition in visual cortical neurons. *Nature*, 6683(393):369–73, 1998.

[6] E. Capobianco. Independent multiresolution component analysis and matching pursuit. *Comput. Stat. Data Anal.*, 42(3):385–402, 2003. ISSN 0167-9473.

[7] S. Celebrini, S. Thorpe, Y. Trotter, and M. Imbert. Dynamics of orientation coding in area V1 of the awake primate. *Vis Neurosci*, 5(10):811–25, 1993.

[8] G. Davis. *Adaptive Nonlinear Approximations.* PhD thesis, New York University, 1994.

[9] P. Durka, D. Ircha, and K. J. Blinowska. Stochastic time-frequency dictionaries for matching pursuit. *IEEE Tran Signal Process*, 49(3):507–510, March 2001.

[10] A. Grinvald, E. Lieke, R. Frostig, and R. Hildesheim. Cortical point-spread function and long-range lateral interactions revealed by real-time optical imaging of macaque monkey primary visual cortex. *The Journal of Neuroscience*, 14(5): 2545–68, May 1994.

[11] J. P. Jones and L. A. Palmer. An evaluation of the two-dimensional gabor filter model of simple receptive fields in cat striate cortex. *J. Neurophysiol.*, 58(6): 1233–58, 1987.

[12] L. Lapicque. Recherches quantitatives sur l'excitation électrique des nerfs traitée comme une polarisation. *J. Physiol. (Paris)*, 9:620–35, 1907.

[13] D. MacKay. *Information Theory, Inference and Learning Algorithms*. Cambridge University Press, 2003.

[14] S. Mallat. *A wavelet tour of signal processing*. Academic Press, 1998.

[15] S. Mallat and Z. Zhang. Matching pursuit with time-frequency dictionaries. *IEEE Transactions on Signal Processing*, 41(12):3397–3414, 1993.

[16] B. Olshausen and D. Field. Learning a sparse code for natural images produces a multiscale family of localized receptive fields. *Nature*, 1996.

[17] B. Olshausen and D. Field. Sparse coding with an overcomplete basis set: A strategy employed by V1? *Vision Research*, 37:3311–25, 1998.

[18] B. Olshausen. What is the other 85% of V1 doing? In L. van Hemmen T.J. Sejnowski, editor, *Problems in Systems Neuroscience*. Oxford University Press, 2004.

[19] A. Pece. The problem of sparse spike coding. Technical Report DIKU-TR-2001-02, Institue of computer science, University of Copenhagen, Copenhagen, Denmark, 2001.

[20] L. Perrinet. Finding independent components using spikes : a natural result of hebbian learning in a sparse spike coding scheme. *Natural Computing*, 3(2):159–75, January 2004. URL `http://laurent.perrinet.free.fr/publi/perrinet03nc.pdf`.

[21] L. Perrinet, M. Samuelides, and S. Thorpe. Coding static natural images using spiking event times : do neurons cooperate? *IEEE Transactions on Neural Networks, Special Issue on 'Temporal Coding for Neural Information Processing'*, 15(5):1164– 1175, September 2004. ISSN 1045-9227. URL `http://laurent.perrinet.free.fr/publi/perrinet03ieee.pdf`.

[22] C. Poynton. Gamma and its disguises. *Journal of the Society of Motion Picture and Television Engineers*, 102(12):1099–108, December 1993.

[23] D. Ringach. Spatial structure and symmetry of simple-cell receptive fields in macaque primary visual cortex. *J. Neurophysiology*, 88:455–63, 2002.

[24] D. Ringach, R. Shapley, and M. Hawken. Orientation selectivity in macaque V1: diversity and laminar dependence. *J. Neurosci.*, 22(13):5639–51, 2002.

[25] F. Rosenblatt. Perceptron simulation experiments. *Proceedings of the I. R. E.*, 20: 167–192, 1960.

[26] O. Schwartz and E. Simoncelli. Natural signal statistics and sensory gain control. *Nature Neuroscience*, 4(8):819–25, 2001.

[27] C. Shannon and W. Weaver. *The mathematical theory of communication.* The University of Illinois Press, Urbana, 1964.

[28] E. Simoncelli and B. Olshausen. Natural image statistics and neural representation. *Annual Review of Neuroscience*, 24:1193–216, 2001.